Gregory Chan

EECS 339 Machine Learning HW1

Problem 1

Candidate-Elimination with JapaneseEconomyCar as target function
Training Examples

Sample Number	Origin	Manufacturer	Color	Decade	Type	Classification
1	Japan	Honda	Blue	1980	Economy	1
2	Japan	Toyota	Green	1970	Sports	0
3	Japan	Toyota	Blue	1990	Economy	1
4	USA	Chrysler	Red	1980	Economy	0
5	Japan	Honda	White	1980	Economy	1

Candidate Elimination Trace

Loop Iteration	Specific	General
0 (initial setup)	{<ø,ø,ø,ø,ø,ø>}	{(?,?,?,?,?,?)}
1	{(Japan, Honda, Blue, 1980, Economy)}	{(?,?,?,?,?,?)}
2	{(Japan, Honda, Blue, 1980, Economy)}	{(?, Honda, ?, ?, ?), (?, ?, Blue, ?, ?), (?, ?, ?, 1980, ?), (?, ?, ?, ?, Economy)}
3	{(Japan, ?, Blue, ?, Economy)}	{(?, ?, Blue, ?, ?), (?, ?, ?, ?, Economy)}
4	{(Japan, ?, Blue, ?, Economy)}	{(?, ?, Blue, ?, ?), (Japan, ?, ?, ?, Economy)}
5	{(Japan, ?, ?, ?, Economy)}	{(Japan, ?, ?, ?, Economy)}

specific = general, both singleton therefore, we are done.

Problem 2

Part A

Define a distance metric for two Facebook users.

A measurement is a metric if it satisfies reflexivity, symmetry, non-negative, and the triangle inequality. A measurement we can use to determine the distance between two Facebook users is their mutual friends. The distance between two Facebook users can then be expressed as

Part B

For a population of varying ages from children to the elderly, I would expect the range for height would be from 2’ to 6’6”, for weight it would be from 25lb to 300lb, and for the number of hairs on a persons head it might vary from 0 to 150,000.

We should not weight the attributes equally. When considering age grouping, it would make sense to normalize the data and then it might also make some sense to weight the height and the number of hairs on a person's head because it will more easily differentiate children, teenagers, and middle-aged adults, and the elderly. Weight, on the other hand, may result in a misclassification of a heavy teenager as an adult or a light adult as a teenager, and seeing as weight varies less between boundaries between teenagers, adults and the elderly, it would make sense to me to not weight this parameter as much.

We can design a metric by using normalization and weighting the features, next we can run a clustering algorithm and adjust weights to improve the performance of our algorithm.

Part C
We should be able to use this without much modification. DNA is represented as adenine (A), guanine (G), cytosine (C), and thymine (T). We can therefore represent a short strand of DNA as so: AAGGCTTGGAA. The problem of determining the distance between two strands of DNA can be determined using a modified measure as a metric.

One way we could do this is by aligning two strings of DNA, and inserting spaces into the strings and calculating a score for each position. If neither string contains a space and the string values are the same at an index, then we can assign it -1. If neither string contains a space and the string values are not the same, then we can assign it a score of 1. Finally, we can assign a score of 2 if either has a space. The distance between the strings is then simply the distance between the score achieved after processing the two strings.

Problem 3
Part A
The size of D_T=|L|^|T|. This is because the number of training sets is |T|, but each example in the training set can only be labeled by one value, thus, D_T=|L|^|T|

Part B
The probability of the learner finding a hypothesis h that is indistinguishable from the target function f is 1/(2^100) because the size of the largest distinguished hypothesis is 2^100, and to distinguish one of them given no other factors should be 1/(2^100)

Part C
The probability that hypothesis h that is indistinguishable from the target function, given X is 1/(2^100) because give |x| and |T| and the |T| examples are consistent with hypothesis h and since the remaining |x| - |T| = 100 examples consistent with f, the size of the largest set of distinguished hypothesis is 2^(|x|-|T|)=2^100. Since we're calculating the probability of one being distinguished, we end up with 1/(2^100).

Problem 4
Part A
I wasn't able to complete the code for problem 6.

Part B
I wasn't able to complete the code for problem 6.

Problem 5
Part A

=== Run information ===

Scheme:weka.classifiers.trees.Id3
Relation: IvyLeague copy
Instances: 62
Attributes: 8
GoodGrades
GoodLetters
GoodSAT
IsRich
HasScholarship
ParentAlum
SchoolActivities
CLASS
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

Id3

IsRich = true
| GoodLetters = true
| | GoodGrades = true : true
| | GoodGrades = false
| | | GoodSAT = true : true
| | | GoodSAT = false : false
| GoodLetters = false
| | GoodGrades = true
| | | SchoolActivities = true : true
| | | SchoolActivities = false : false
| | GoodGrades = false : false
IsRich = false
| HasScholarship = true
| | GoodSAT = true
| | | GoodLetters = true : true
| | | GoodLetters = false : false
| | GoodSAT = false : false
| HasScholarship = false : false

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances 60 96.7742 %

Incorrectly Classified Instances 2 3.2258 %
Kappa statistic 0.9355
Mean absolute error 0.0323
Root mean squared error 0.1796
Relative absolute error 6.448 %
Root relative squared error 35.8995 %
Total Number of Instances 62

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	1	0.065	0.939	1	0.969	0.968	true
	0.935	0	1	0.935	0.967	0.968	false
Weighted Avg.	0.968	0.032	0.97	0.968	0.968	0.968	

=== Confusion Matrix ===

a b <-- classified as
31 0 | a = true
2 29 | b = false

A text representation of the tree is provided above.

A confusion matrix is a matrix where each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. So for the provided confusion matrix, the number of instances that should be classified as ‘a’ is 31 and the number of instances that have been correctly classified as ‘a’, and 0 instance have been classified as ‘b’. Likewise, the number instances that should be classified as ‘b’ is 29, but there are two instances that are incorrectly classified as ‘a’.

Part B

```
IsRich = true
|  GoodLetters = true : true (25.0/1.0)
|  GoodLetters = false
|  |  GoodGrades = true
|  |  |  SchoolActivities = true : true (3.0)
|  |  |  SchoolActivities = false : false (4.0)
|  |  GoodGrades = false : false (6.0)
IsRich = false
|  HasScholarship = true
|  |  GoodSAT = true : true (5.0/1.0)
|  |  GoodSAT = false : false (6.0)
|  HasScholarship = false : false (13.0)
```

confusion matrix:
a b <-- classified as
30 1 | a = true
2 29 | b = false

The C4.5 algorithm with pruning did not outperform the straightforward ID3 algorithm because the correctness of the C4.5 with pruning is less than the straightforward ID3. This result might be as a result of the size of the data set because there is very little data, so pruning this data could largely impact the correctness. Thus, it might not outperform the straightforward ID3

Part C
The performance of the algorithms on these two data sets is very different. The accuracy of MajorityRule.txt is not very good, and the decision tree for MajorityRule.txt is also much more complex.

The ID3 prefers shorter tress to taller tress, which means that it is more inclined to picking the classification most similar to the parent node. However, for the structure of the concepts for the MajorityRule data, it seems that the tree is very complex with many more nodes. This may be an overfitting issue.

Part D
Reduce error pruning didn't seem to have a positive impact on the performance when measured with MajorityRule because the it is a small data set. Thus, pruning the will not impact the result in a significant way.

Problem 6
I wasn't able to complete all of the problem 6 code. Attached is the code I had started working on.